Using Monolingual Data in Neural Machine Translation: a Systematic Study

Franck Burlot
Lingua Custodia
1, Place Charles de Gaulle
78180 Montigny-le-Bretonneux
franck.burlot@linguacustodia.com

François Yvon
LIMSI, CNRS, Université Paris Saclay
Campus Universitaire d'Orsay
F-91 403 Orsay Cédex
francois.yvon@limsi.fr

Abstract

Neural Machine Translation (MT) has radically changed the way systems are developed. A major difference with the previous generation (Phrase-Based MT) is the way monolingual target data, which often abounds, is used in these two paradigms. While Phrase-Based MT can seamlessly integrate very large language models trained on billions of sentences, the best option for Neural MT developers seems to be the generation of artificial parallel data through *back-translation* - a technique that fails to fully take advantage of existing datasets. In this paper, we conduct a systematic study of back-translation, comparing alternative uses of monolingual data, as well as multiple data generation procedures. Our findings confirm that back-translation is very effective and give new explanations as to why this is the case. We also introduce new data simulation techniques that are almost as effective, yet much cheaper to implement.

1 Introduction

The new generation of Neural Machine Translation (NMT) systems is known to be extremely data hungry (Koehn and Knowles, 2017). Yet, most existing NMT training pipelines fail to fully take advantage of the very large volume of monolingual source and/or parallel data that is often available. Making a better use of data is particularly critical in domain adaptation scenarios, where parallel adaptation data is usually assumed to be small in comparison to out-of-domain parallel data, or to in-domain monolingual texts. This situation sharply contrasts with the previous generation of statistical MT engines (Koehn, 2010), which could seamlessly integrate very large amounts of non-parallel documents, usually with a large positive effect on translation quality.

Such observations have been made repeatedly and have led to many innovative techniques to integrate monolingual data in NMT, that we review shortly. The most successful approach to date is the proposal of Sennrich et al. (2016a), who use monolingual target texts to generate artificial parallel data via backward translation (BT). This technique has since proven effective in many subsequent studies. It is however very computationally costly, typically requiring to translate large sets of data. Determining the "right" amount (and quality) of BT data is another open issue, but we observe that experiments reported in the literature only use a subset of the available monolingual resources. This suggests that standard recipes for BT might be sub-optimal.

This paper aims to better understand the strengths and weaknesses of BT and to design more principled techniques to improve its effects. More specifically, we seek to answer the following questions: since there are many ways to generate pseudo parallel corpora, how important is the quality of this data for MT performance? Which properties of back-translated sentences actually matter for MT quality? Does BT act as some kind of regularizer (Domhan and Hieber, 2017)? Can BT be efficiently simulated? Does BT data play the same role as a target-side language modeling, or are they complementary? BT is often used for domain adaptation: can the effect of having more in-domain data be sorted out from the mere increase of training material (Sennrich et al., 2016a)? For studies related to the impact of varying the size of BT data, we refer the readers to the recent work of Poncelas et al. (2018).

To answer these questions, we have reimplemented several strategies to use monolingual data in NMT and have run experiments on two language pairs in a very controlled setting (see § 2). Our main results (see § 4 and § 5) suggest promising directions for efficient domain adaptation with cheaper techniques than conventional BT.

Proceedings of the Third Conference on Machine Translation (WMT), Volume 1: Research Papers, pages 144–155

https://doi.org/10.18653/v1/W18-64015

	Out-of-domain		In-domain	
	Sents	Token	Sents	Token
en-fr	4.0M	86.8M/97.8M	1.9M	46.0M/50.6M
en-de	4.1M	84.5M/77.8M	1.8M	45.5M/43.4M

Table 1: Size of parallel corpora

2 Experimental Setup

2.1 In-domain and out-of-domain data

We are mostly interested with the following training scenario: a large out-of-domain parallel corpus, and limited monolingual in-domain data. We focus here on the *Europarl* domain, for which we have ample data in several languages, and use as in-domain training data the Europarl corpus[1] (Koehn, 2005) for two translation directions: English→German and English→French. As we study the benefits of monolingual data, most of our experiments only use the target side of this corpus. The rationale for choosing this domain is to (i) to perform large scale comparisons of synthetic and natural parallel corpora; (ii) to study the effect of BT in a well-defined domain-adaptation scenario. For both language pairs, we use the Europarl tests from 2007 and 2008[2] for evaluation purposes, keeping test 2006 for development. When measuring out-of-domain performance, we will use the WMT newstest 2014.

2.2 NMT setups and performance

Our baseline NMT system implements the attentional encoder-decoder approach (Cho et al., 2014; Bahdanau et al., 2015) as implemented in Nematus (Sennrich et al., 2017) on 4 million out-of-domain parallel sentences. For French we use samples from News-Commentary-11 and Wikipedia from WMT 2014 shared translation task, as well as the Multi-UN (Eisele and Chen, 2010) and EU-Bookshop (Skadiņš et al., 2014) corpora. For German, we use samples from News-Commentary-11, Rapid, Common-Crawl (WMT 2017) and Multi-UN (see table 1). Bilingual BPE units (Sennrich et al., 2016b) are learned with 50k merge operations, yielding vocabularies of about respectively 32k and 36k for English→French and 32k and 44k for English→German.

Both systems use 512-dimensional word embeddings and a single hidden layer with 1024 cells. They are optimized using Adam (Kingma and Ba, 2014) and early stopped according to the validation performance. Training lasted for about three weeks on an Nvidia K80 GPU card.

Systems generating back-translated data are trained using the same out-of-domain corpus, where we simply exchange the source and target sides. They are further documented in § 3.1.

For the sake of comparison, we also train a system that has access to a large batch of in-domain parallel data following the strategy often referred to as "fine-tuning": upon convergence of the baseline model, we resume training with a 2M sentence in-domain corpus mixed with an equal amount of randomly selected out-of-domain natural sentences, with the same architecture and training parameters, running validation every 2000 updates with a patience of 10. Since BPE units are selected based only on the out-of-domain statistics, fine-tuning is performed on sentences that are slightly longer (ie. they contain more units) than for the initial training. This system defines an upper-bound of the translation performance and is denoted below as `natural`.

Our baseline and topline results are in Table 2, where we measure translation performance using BLEU (Papineni et al., 2002), BEER (Stanojević and Sima'an, 2014) (higher is better) and characTER (Wang et al., 2016) (smaller is better). As they are trained from much smaller amounts of data than current systems, these baselines are not quite competitive to today's best system, but still represent serious baselines for these datasets. Given our setups, fine-tuning with in-domain natural data improves BLEU by almost 4 points for both translation directions on in-domain tests; it also improves, albeit by a smaller margin, the BLEU score of the out-of-domain tests.

3 Using artificial parallel data in NMT

A simple way to use monolingual data in MT is to turn it into synthetic parallel data and let the training procedure run as usual (Bojar and Tamchyna, 2011). In this section, we explore various ways to implement this strategy. We first reproduce results of Sennrich et al. (2016a) with BT of various qualities, that we then analyze thoroughly.

3.1 The quality of Back-Translation

3.1.1 Setups

BT requires the availability of an MT system in the reverse translation direction. We consider here

[1] Version 7, see `www.statmt.org/europarl/`.

[2] `www.statmt.org/wmt08`.

English→French									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	31.25	62.14	51.89	32.17	62.35	50.79	33.06	61.97	48.56
backtrans-bad	31.55	62.39	51.50	31.89	62.23	51.73	31.99	61.59	48.86
backtrans-good	32.99	63.43	49.58	33.25	63.08	49.29	33.52	62.62	47.23
backtrans-nmt	33.30	63.33	50.02	33.39	63.09	49.48	34.11	62.76	46.94
fwdtrans-nmt	31.93	62.55	50.84	32.62	62.66	49.83	33.56	62.44	47.65
backfwdtrans-nmt	33.09	63.19	50.08	33.70	63.25	48.83	34.00	62.76	47.22
natural	35.10	64.71	48.33	35.29	64.52	48.26	34.96	63.08	46.67
English→German									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	21.36	57.08	63.32	21.27	57.11	60.67	22.49	57.79	55.64
backtrans-bad	21.84	57.85	61.24	21.04	57.44	59.77	22.28	57.70	55.49
backtrans-good	23.33	59.03	58.84	23.11	57.14	57.14	22.87	58.09	54.91
backtrans-nmt	23.00	59.12	58.31	23.10	58.85	56.67	22.91	58.12	54.67
fwdtrans-nmt	21.97	57.46	61.99	21.89	57.53	59.71	22.52	57.93	55.13
backfwdtrans-nmt	22.99	58.37	60.45	22.82	58.14	58.80	23.04	58.17	54.96
natural	26.74	61.14	56.19	26.16	60.64	54.76	23.84	58.64	54.23

Table 2: Performance *wrt.* different BT qualities

	French→English				German→English			
	test-07	test-08	nt-14	unk	test-07	test-08	nt-14	unk
backtrans-bad	18.86	19.27	20.49	3.22%	14.66	14.62	15.07	1.45%
backtrans-good	29.71	29.51	32.10	0.24%	24.19	24.19	25.75	0.73%
backtrans-nmt	31.10	31.43	31.27	0.0%	26.02	26.03	26.98	0.0%

Table 3: BLEU scores for (backward) translation into English

three MT systems of increasing quality:

1. backtrans-bad: this is a very poor SMT system trained using only 50k parallel sentences from the out-of-domain data, and no additional monolingual data. For this system as for the next one, we use Moses (Koehn et al., 2007) out-of-the-box, computing alignments with Fastalign (Dyer et al., 2013), with a minimal pre-processing (basic tokenization). This setting provides us with a pessimistic estimate of what we could get in low-resource conditions.

2. backtrans-good: these are much larger SMT systems, which use the same parallel data as the baseline NMTs (see § 2.2) and all the English monolingual data available for the WMT 2017 shared tasks, totalling approximately 174M sentences. These systems are strong, yet relatively cheap to build.

3. backtrans-nmt: these are the best NMT systems we could train, using settings that replicate the forward translation NMTs.

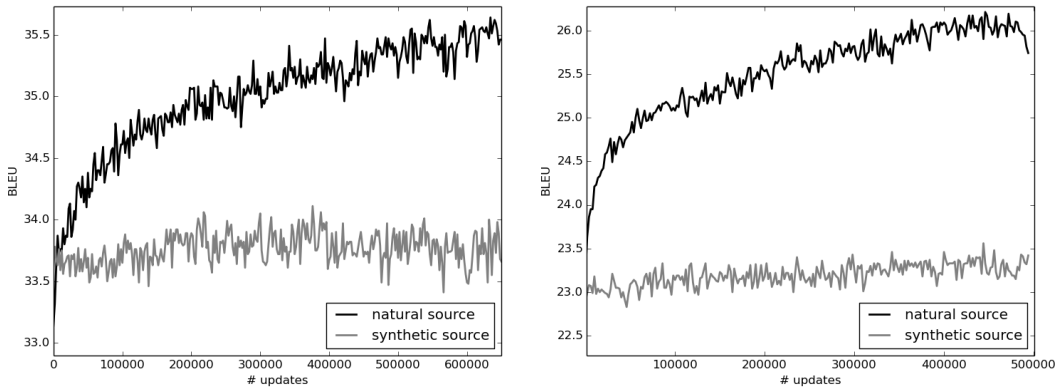
Note that we do not use any in-domain (*Europarl*) data to train these systems. Their performance is reported in Table 3, where we observe a 12 BLEU points gap between the worst and best systems (for both languages).

As noted *eg.* in (Park et al., 2017; Crego and Senellart, 2016), artificial parallel data obtained through *forward-translation* (FT) can also prove advantageous and we also consider a FT system (fwdtrans-nmt): in this case the *target* side of the corpus is artificial and is generated using the baseline NMT applied to a natural source.

3.1.2 BT quality does matter

Our results (see Table 2) replicate the findings of (Sennrich et al., 2016a): large gains can be obtained from BT (nearly +2 BLEU in French and German); better artificial data yields better translation systems. Interestingly, our best Moses system is almost as good as the NMT and an order of magnitude faster to train. Improvements obtained with the bad system are much smaller; contrary to the better MTs, this system is even detrimental for the out-of-domain test.

Gains with forward translation are significant, as in (Chinea-Rios et al., 2017), albeit about half as good as with BT, and result in small improvements for the in-domain and for the out-of-domain tests. Experiments combining forward and backward translation (backfwdtrans-nmt), each



English→French English→German

Figure 1: Learning curves from `backtrans-nmt` and `natural`. Artificial parallel data is more prone to overfitting than natural data.

using a half of the available artificial data, do not outperform the best BT results.

We finally note the large remaining difference between BT data and natural data, even though they only differ in their source side. This shows that at least in our domain-adaptation settings, BT does not really act as a regularizer, contrarily to the findings of (Poncelas et al., 2018; Sennrich et al., 2016b). Figure 3.1.1 displays the learning curves of these two systems. We observe that `backtrans-nmt` improves quickly in the earliest updates and then stays horizontal, whereas `natural` continues improving, even after 400k updates. Therefore BT does not help to avoid overfitting, it actually encourages it, which may be due "easier" training examples (cf. § 3.2).

3.2 Properties of back-translated data

Comparing the natural and artificial sources of our parallel data *wrt.* several linguistic and distributional properties, we observe that (see Fig. 2 - 3):

(i) artificial sources are on average shorter than natural ones: when using BT, cases where the source is shorter than the target are rarer; cases when they have the same length are more frequent.

(ii) automatic word alignments between artificial sources tend to be more monotonic than when using natural sources, as measured by the average Kendall τ of source-target alignments (Birch and Osborne, 2010): for French-English the respective numbers are 0.048 (natural) and 0.018 (artificial); for German-English 0.068 and 0.053. Using more monotonic sentence pairs turns out to be a facilitating factor for NMT, as also noted by Crego and Senellart (2016).

(iii) syntactically, artificial sources are simpler than real data; We observe significant differences in the distributions of tree depths.[3]

(iv) distributionally, plain word occurrences in artificial sources are more concentrated; this also translates into both a slower increase of the number of types *wrt.* the number of sentences and a smaller number of rare events.

The intuition is that properties (i) and (ii) should help translation as compared to natural source, while property (iv) should be detrimental. We checked (ii) by building systems with only 10M words from the natural parallel data selecting these data either randomly or based on the regularity of their word alignments. Results in Table 4 show that the latter is much preferable for the overall performance. This might explain that the mostly monotonic BT from Moses are almost as good as the fluid BT from NMT and that both boost the baseline.

4 Stupid Back-Translation

We now analyze the effect of using much simpler data generation schemes, which do not require the availability of a backward translation engine.

[3]Parses were automatically computed with CoreNLP (Manning et al., 2014).

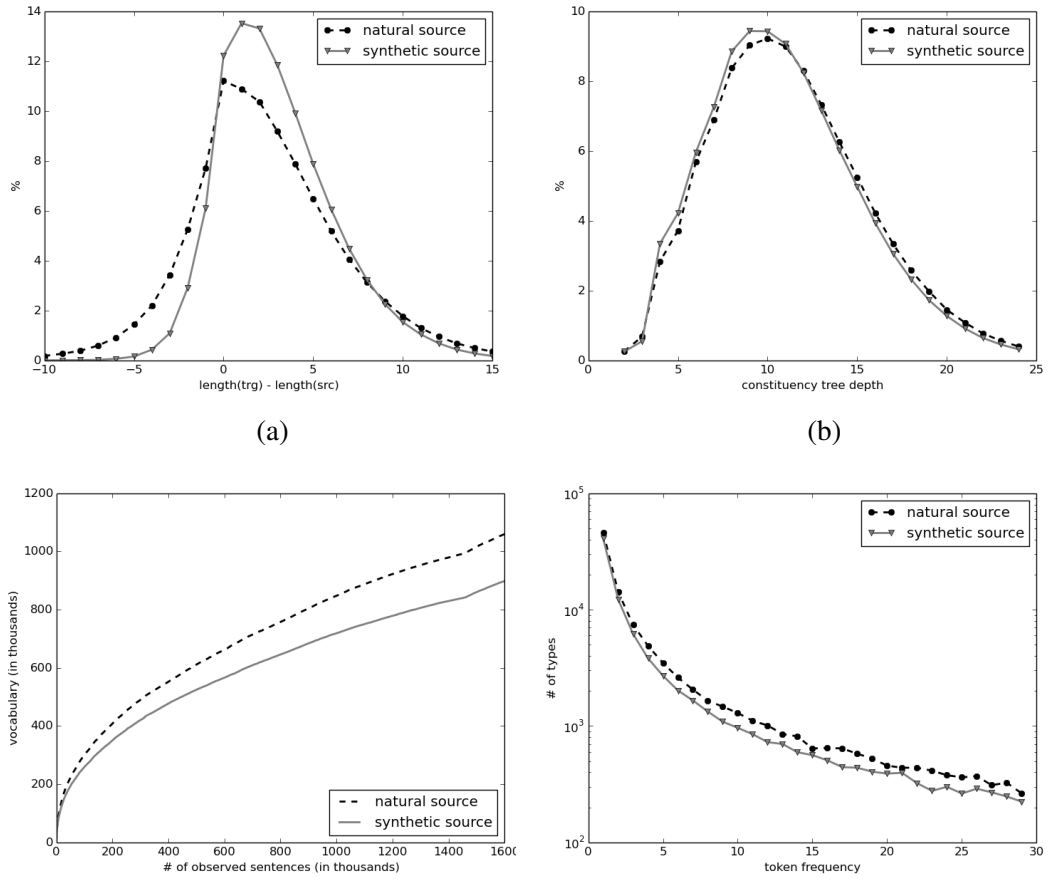


Figure 2: Properties of pseudo-English data obtained with `backtrans-nmt` from French. The synthetic source contains shorter sentences (a) and slightly simpler syntax (b). The vocabulary growth *wrt.* an increasing number of observed sentences (c) and the token-type correlation (d) suggest that the natural source is lexically richer.

	test-07			**test-08**			**newstest-14**		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
random	32.08	62.98	50.78	32.66	62.86	49.99	23.05	55.38	58.51
monotonic	33.52	63.75	49.51	33.73	63.59	48.91	32.16	61.75	48.64

Table 4: Selection strategies for BT data (English-French)

4.1 Setups

We use the following cheap ways to generate pseudo-source texts:

1. `copy`: in this setting, the source side is a mere copy of the target-side data. Since the source vocabulary of the NMT is fixed, copying the target sentences can cause the occurrence of OOVs. To avoid this situation, Currey et al. (2017) decompose the target words into source-side units to make the copy look like source sentences. Each OOV found in the copy is split into smaller units until all the resulting chunks are in the source vocabulary.

2. `copy-marked`: another way to integrate copies without having to deal with OOVs is to augment the source vocabulary with a copy of the target vocabulary. In this setup, Ha et al. (2016) ensure that both vocabularies never overlap by marking the target word copies with a special language identifier. Therefore the English word *resume* cannot be confused with the homographic French word, which is marked *@fr@resume*.

3. `copy-dummies`: instead of using actual copies, we replace each word with "dummy" tokens. We use this unrealistic setup to observe the training over noisy and hardly informative source sentences.

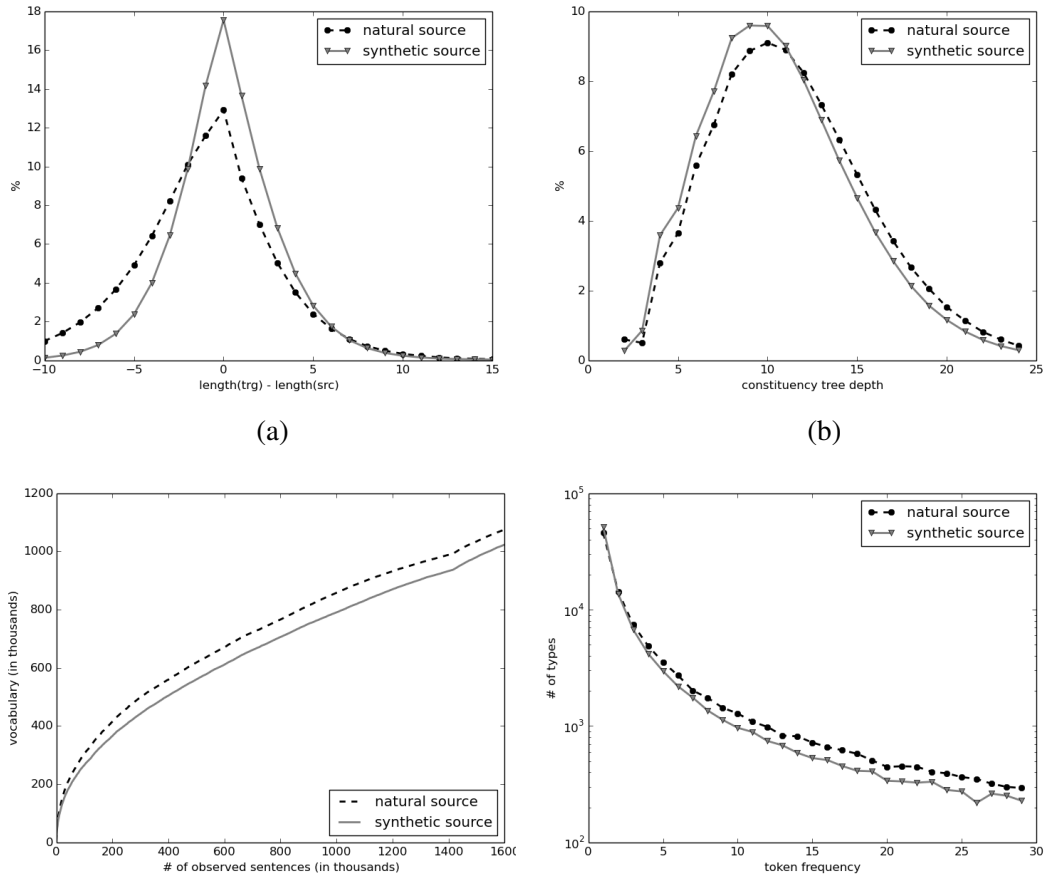


Figure 3: Properties of pseudo-English data obtained with `backtrans-nmt` (back-translated from German). Tendencies similar to English-French can be observed and difference in syntax complexity is even more visible.

We then use the procedures described in § 2.2, except that the pseudo-source embeddings in the `copy-marked` setup are pretrained for three epochs on the in-domain data, while all remaining parameters are frozen. This prevents random parameters from hurting the already trained model.

4.2 Copy+marking+noise is not so stupid

We observe that the `copy` setup has only a small impact on the English-French system, for which the baseline is already strong. This is less true for English-German where simple copies yield a significant improvement. Performance drops for both language pairs in the `copy-dummies` setup.

We achieve our best gains with the `copy-marked` setup, which is the best way to use a copy of the target (although the performance on the out-of-domain tests is at most the same as the baseline). Such gains may look surprising, since the NMT model does not need to learn to translate but only to copy the source. This is indeed what happens: to confirm this, we built a fake test set having identical source and target side (in French). The average cross-entropy for this test set is 0.33, very close to 0, to be compared with an average cost of 58.52 when we process an actual source (in English). This means that the model has learned to copy words from source to target with no difficulty, even for sentences not seen in training. A follow-up question is whether training a copying task instead of a translation task limits the improvement: would the NMT learn better if the task was harder? To measure this, we introduce noise in the target sentences copied onto the source, following the procedure of Lample et al. (2017): it deletes random words and performs a small random permutation of the remaining words. Results (+ *Source noise*) show no difference for the French in-domain test sets, but bring the out-of-domain score to the level of the baseline. Finally, we observe a significant improvement on German in-domain

English→French									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	31.25	62.14	51.89	32.17	62.35	50.79	33.06	61.97	48.56
`copy`	31.65	62.45	52.09	32.23	62.37	52.20	32.80	61.99	49.05
`copy-dummies`	30.89	62.06	52.07	31.51	61.98	51.46	31.43	60.92	50.58
`copy-marked`	32.01	62.66	51.57	32.31	62.52	51.46	32.33	61.55	49.44
+ Source noise	31.87	62.52	52.69	32.64	62.55	51.63	33.04	62.11	48.47
English→German									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	21.36	57.08	63.32	21.27	57.11	60.67	22.49	57.79	55.64
`copy`	22.15	57.95	61.49	21.95	57.72	59.58	22.59	57.83	55.44
`copy-dummies`	21.73	57.84	61.35	21.38	57.38	60.10	21.12	56.81	57.21
`copy-marked`	22.58	58.23	61.10	22.47	57.97	59.24	22.53	57.54	55.85
+ Source noise	22.92	58.62	60.27	22.83	58.36	58.48	22.34	57.47	55.72

Table 5: Performance *wrt.* various stupid BTs

test sets, compared to the baseline (about +1.5 BLEU). This last setup is even almost as good as the `backtrans-nmt` condition (see § 3.1) for German. This shows that learning to reorder and predict missing words can more effectively serve our purposes than simply learning to copy.

5 Towards more natural pseudo-sources

Integrating monolingual data into NMT can be as easy as copying the target into the source, which already gives some improvement; adding noise makes things even better. We now study ways to make pseudo-sources look more like natural data, using the framework of Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), an idea borrowed from Lample et al. (2017)[4].

5.1 GAN setups

In our setups, we use a marked target copy, viewed as a *fake* source, which a *generator* encodes so as to fool a discriminator *trained* to distinguish a *fake* from a *natural* source. Our architecture contains two distinct encoders, one for the natural source and one for the pseudo-source. The latter acts as the generator (G) in the GAN framework, computing a representation of the pseudo-source that is then input to a discriminator (D), which has to sort natural from artificial encodings. D assigns a probability of a sentence being natural.

During training, the cost of the discriminator is computed over two batches, one with natural (out-of-domain) sentences $\mathbf{x}$ and one with (in-domain) pseudo-sentences $\mathbf{x}'$. The discriminator is a bidirectional-Recurrent Neural Network (RNN) of dimension 1024. Averaged states are passed to a single feed-forward layer, to which a sigmoid is applied. It inputs encodings of natural ($E(\mathbf{x})$) and pseudo-sentences ($G(\mathbf{x}')$) and is trained to optimize:

$$J^{(D)} = -\frac{1}{2}\mathbb{E}_{\mathbf{x}\sim p_{\text{real}}} \log D(E(\mathbf{x})) - \frac{1}{2}\mathbb{E}_{\mathbf{x}'\sim p_{\text{pseudo}}} \log(1 - D(G(\mathbf{x}')))$$

G's parameters are updated to maximally fool D, thus the loss $J^{(G)}$:

$$J^{(G)} = -\mathbb{E}_{\mathbf{x}'\sim p_{\text{pseudo}}} \log D(G(\mathbf{x}'))$$

Finally, we keep the usual MT objective. ($\mathbf{s}$ is a real or pseudo-sentence):

$$J^{(\text{MT})} = \log p(\mathbf{y}|\mathbf{s}) = -\mathbb{E}_{\mathbf{s}\sim p_{\text{all}}} \log \text{MT}(\mathbf{s})$$

We thus need to train three sets of parameters: $\theta^{(D)}, \theta^{(G)}$ and $\theta^{(\text{MT})}$ (MT parameters), with $\theta^{(G)} \in \theta^{(\text{MT})}$. The pseudo-source encoder and embeddings are updated *wrt.* both $J^{(G)}$ and $J^{(\text{MT})}$. Following (Goyal et al., 2016), $\theta^{(G)}$ is updated only when D's accuracy exceeds 75%. On the other hand, $\theta^{(D)}$ is not updated when its accuracy exceeds 99%. At each update, two batches are generated for each type of data, which are encoded with the real or pseudo-encoder. The encoder outputs serve to compute $J^{(D)}$ and $J^{(G)}$. Finally, the pseudo-source is encoded again (once G is updated), both encoders are plugged into the translation model and the MT cost is back-propagated down to the real and pseudo-word embeddings. Pseudo-encoder and discriminator parameters are pre-trained for 10k updates. At test time, the pseudo-encoder is ignored and inference is run as usual.

[4] Our implementation is available at `https://github.com/franckbrl/nmt-pseudo-source-discriminator`

English→French									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	31.25	62.14	51.89	32.17	62.35	**50.79**	**33.06**	61.97	48.56
`copy-marked`	32.01	62.66	**51.57**	32.31	62.52	51.46	32.33	61.55	49.44
+ GANs	31.95	62.55	52.87	32.24	62.47	52.16	32.86	61.90	48.97
`copy-marked` + noise	31.87	62.52	52.69	32.64	62.55	51.63	33.04	62.11	48.47
+ GANs	**32.41**	**62.78**	52.25	**32.79**	**62.72**	50.92	33.01	**61.98**	**48.37**
`backtrans-nmt`	**33.30**	**63.33**	**50.02**	**33.39**	**63.09**	**49.48**	**34.11**	**62.76**	**46.94**
+ Distinct encoders	32.29	62.83	51.55	32.98	62.91	51.19	33.60	62.43	48.06
+ GANs	32.91	63.08	51.17	33.24	62.93	50.82	33.77	62.42	47.80
`natural`	35.10	64.71	48.33	35.29	64.52	48.26	34.96	63.08	46.67
English→German									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	21.36	57.08	63.32	21.27	57.11	60.67	22.49	**57.79**	55.64
`copy-marked`	22.58	58.23	61.10	22.47	57.97	59.24	22.53	57.54	55.85
+ GANs	22.71	58.25	61.25	22.44	57.86	59.28	**22.81**	57.54	55.99
`copy-marked` + noise	22.92	58.62	60.27	**22.83**	**58.36**	**58.48**	22.34	57.47	55.72
+ GANs	**23.01**	**58.66**	**60.22**	22.53	58.16	58.65	22.64	57.70	**55.48**
`backtrans-nmt`	23.00	59.12	**58.31**	23.10	58.85	**56.67**	22.91	58.12	**54.67**
+ Distinct encoders	23.62	58.83	59.74	23.10	**58.50**	58.19	22.82	**57.91**	54.96
+ GANs	**23.65**	**58.85**	59.70	**23.20**	**58.50**	58.22	**23.00**	57.89	55.15
`natural`	26.74	61.14	56.19	26.16	60.64	54.76	23.84	58.64	54.23

Table 6: Performance *wrt.* different GAN setups

English→French									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	31.25	62.14	**51.89**	32.17	62.35	**50.79**	33.06	61.97	48.56
`deep-fusion`	31.85	62.52	52.27	32.25	62.40	51.64	**33.65**	**62.40**	**48.24**
`copy-marked` + noise + GANs	**32.41**	**62.78**	52.25	**32.79**	**62.72**	50.92	33.01	61.98	48.37
+`deep-fusion`	31.96	62.59	51.96	32.59	62.59	51.65	32.96	61.95	48.95
English→German									
	test-07			test-08			newstest-14		
	BLEU	BEER	CTER	BLEU	BEER	CTER	BLEU	BEER	CTER
Baseline	21.36	57.08	63.32	21.27	57.11	60.67	22.49	57.79	55.64
`deep-fusion`	21.65	57.57	62.38	21.33	57.33	60.54	**23.10**	**58.06**	**55.33**
`copy-marked` + noise + GANs	**23.01**	**58.66**	**60.22**	**22.53**	**58.16**	**58.65**	22.64	57.70	55.48
+`deep-fusion`	23.07	58.50	60.47	22.86	58.18	58.76	22.64	57.46	55.85

Table 7: Deep-fusion model

5.2 GANs can help

Results are in Table 6, assuming the same fine-tuning procedure as above. On top of the `copy-marked` setup, our GANs do not provide any improvement in both language pairs, with the exception of a small improvement for English-French on the out-of-domain test, which we understand as a sign that the model is more robust to domain variations, just like when adding pseudo-source noise. When combined with noise, the French model yields the best performance we could obtain with stupid BT on the in-domain tests, at least in terms of BLEU and BEER. On the News domain, we remain close to the baseline level, with slight improvements in German.

A first observation is that this method brings stupid BT models closer to conventional BT, at a greatly reduced computational cost. While French still remains 0.4 to 1.0 BLEU below very good backtranslation, both approaches are in the same ballpark for German - may be because BTs are better for the former system than for the latter.

Finally note that the GAN architecture has two differences with basic `copy-marked`: (a) a distinct encoder for real and pseudo-sentence; (b) a different training regime for these encoders. To sort out the effects of (a) and (b), we reproduce the GAN setup with BT sentences, instead of copies. Using a separate encoder for the pseudo-source in the `backtrans-nmt` setup can be detrimental to performance (see Table 6): translation degrades in French for all metrics. Adding GANs on top of the pseudo-encoder was not able to make up for the degradation observed in French, but al-

lowed the German system to slightly outperform backtrans-nmt. Even though this setup is unrealistic and overly costly, it shows that GANs are actually helping even good systems.

6 Using Target Language Models

In this section, we compare the previous methods with the use of a target side Language Model (LM). Several proposals exist in the literature to integrate LMs in NMT: for instance, Domhan and Hieber (2017) strengthen the decoder by integrating an extra, source independent, RNN layer in a conventional NMT architecture. Training is performed either with parallel, or monolingual data. In the latter case, word prediction only relies on the source independent part of the network.

6.1 LM Setup

We have followed Gulcehre et al. (2017) and reimplemented[5] their deep-fusion technique. It requires to first independently learn a RNN-LM on the in-domain target data with a cross-entropy objective; then to train the optimal combination of the translation and the language models by adding the hidden state of the RNN-LM as an additional input to the softmax layer of the decoder.

Our RNN-LMs are trained using dl4mt[6] with the target side of the parallel data and the Europarl corpus (about 6M sentences for both French and German), using a one-layer GRU with the same dimension as the MT decoder (1024).

6.2 LM Results

Results are in Table 7. They show that deep-fusion hardly improves the Europarl results, while we obtain about +0.6 BLEU over the baseline on newstest-2014 for both languages. deep-fusion differs from stupid BT in that the model is not directly optimized on the in-domain data, but uses the LM trained on Europarl to maximize the likelihood of the out-of-domain training data. Therefore, no specific improvement is to be expected in terms of domain adaptation, and the performance increases in the more general domain. Combining deep-fusion and copy-marked + noise + GANs brings slight improvements on the German in-domain test sets, and performance out of the domain remains near the baseline level.

7 Re-analyzing the effects of BT

As a follow up of previous discussions, we analyze the effect of BT on the internals of the network. Arguably, using a copy of the target sentence instead of a natural source should not be helpful for the encoder, but is it also the case with a strong BT? What are the effects on the attention model?

7.1 Parameter freezing protocol

To investigate these questions, we run the same fine-tuning using the copy-marked, backtrans-nmt and backtrans-nmt setups. Note that except for the last one, all training scenarios have access to same target training data. We intend to see whether the overall performance of the NMT system degrades when we selectively freeze certain sets of parameters, meaning that they are not updated during fine-tuning.

7.2 Results

BLEU scores are in Table 8. The backtrans-nmt setup is hardly impacted by selective updates: updating the only decoder leads to a degradation of at most 0.2 BLEU. For copy-marked, we were not able to freeze the source embeddings, since these are initialized when fine-tuning begins and therefore need to be trained. We observe that freezing the encoder and/or the attention parameters has no impact on the English-German system, whereas it slightly degrades the English-French one. This suggests that using artificial sources, even of the poorest quality, has a positive impact on all the components of the network, which makes another big difference with the LM integration scenario.

The largest degradation is for natural, where the model is prevented from learning from informative source sentences, which leads to a decrease of 0.4 to over 1.0 BLEU. We assume from these experiments that BT impacts most of all the decoder, and learning to encode a pseudo-source, be it a copy or an actual back-translation, only marginally helps to significantly improve the quality. Finally, in the fwdtrans-nmt setup, freezing the decoder does not seem to harm learning with a natural source.

[5] Our implementation is part of the Nematus toolkit (theano branch): https://github.com/EdinburghNLP/nematus/blob/theano/doc/deep_fusion_lm.md

[6] https://github.com/nyu-dl/dl4mt-tutorial

	English→French			English→German		
	test-07	**test-08**	**nt-14**	**test-07**	**test-08**	**nt-14**
Baseline	31.25	32.17	33.06	21.36	21.27	22.49
`backtrans-nmt`	33.30	33.39	34.11	23.00	23.10	22.91
+ Freeze source embedd.	33.20	33.24	34.16	22.84	22.85	23.00
+ Freeze encoder	33.17	33.25	33.73	22.72	22.74	22.64
+ Freeze attention	33.13	33.22	33.47	23.03	23.01	22.85
`copy-marked`	32.01	32.31	32.33	22.58	22.47	22.53
+ Freeze encoder	31.70	32.39	32.90	22.59	22.30	22.81
+ Freeze attention	31.59	32.39	32.54	22.55	22.13	22.69
`fwdtrans-nmt`	31.93	32.62	33.56	21.97	21.89	22.52
+ Freeze decoder	31.84	32.62	33.35	21.91	21.65	13.61
`natural`	35.10	35.29	34.96	26.74	26.16	23.84
+ Freeze encoder	34.02	34.25	34.09	24.95	25.08	23.44
+ Freeze attention	34.13	34.42	34.19	25.13	24.97	23.35

Table 8: BLEU scores with selective parameter freezing

8 Related work

The literature devoted to the use of monolingual data is large, and quickly expanding. We already alluded to several possible ways to use such data: using back- or forward-translation or using a target language model. The former approach is mostly documented in (Sennrich et al., 2016a), and recently analyzed in (Park et al., 2017), which focus on fully artificial settings as well as pivot-based artificial data; and (Poncelas et al., 2018), which studies the effects of increasing the size of BT data. The studies of Crego and Senellart (2016); Park et al. (2017) also consider forward translation and Chinea-Rios et al. (2017) expand these results to domain adaptation scenarios. Our results are complementary to these earlier studies.

As shown above, many alternatives to BT exist. The most obvious is to use target LMs (Domhan and Hieber, 2017; Gulcehre et al., 2017), as we have also done here; but attempts to improve the encoder using multi-task learning also exist (Zhang and Zong, 2016).

This investigation is also related to recent attempts to consider supplementary data with a valid target side, such as multi-lingual NMT (Firat et al., 2016), where source texts in several languages are fed in the same encoder-decoder architecture, with partial sharing of the layers. This is another realistic scenario where additional resources can be used to selectively improve parts of the model.

Round trip training is another important source of inspiration, as it can be viewed as a way to use BT to perform semi-unsupervised (Cheng et al., 2016) or unsupervised (He et al., 2016) training of NMT. The most convincing attempt to date along these lines has been proposed by Lample et al. (2017), who propose to use GANs to mitigate the difference between artificial and natural data.

9 Conclusion

In this paper, we have analyzed various ways to integrate monolingual data in an NMT framework, focusing on their impact on quality and domain adaptation. While confirming the effectiveness of BT, our study also proposed significantly cheaper ways to improve the baseline performance, using a slightly modified copy of the target, instead of its full BT. When no high quality BT is available, using GANs to make the pseudo-source sentences closer to natural source sentences is an efficient solution for domain adaptation.

To recap our answers to our initial questions: the quality of BT actually matters for NMT (cf. § 3.1) and it seems that, even though artificial source are lexically less diverse and syntactically complex than real sentence, their monotonicity is a facilitating factor. We have studied cheaper alternatives and found out that copies of the target, if properly noised (§ 4), and even better, if used with GANs, could be almost as good as low quality BTs (§ 5): BT is only worth its cost when good BT can be generated. Finally, BT seems preferable to integrating external LM - at least in our data condition (§ 6). Further experiments with larger LMs are needed to confirm this observation, and also to evaluate the complementarity of both strategies. More work is needed to better understand the impact of BT on subparts of the network (§ 7).

In future work, we plan to investigate other cheap ways to generate artificial data. The experimental setup we proposed may also benefit from a refining of the data selection strategies to focus on the most useful monolingual sentences.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the first International Conference on Learning Representations*, San Diego, CA.

Alexandra Birch and Miles Osborne. 2010. LRscore for evaluating lexical and reordering quality in MT. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, WMT '10, pages 327–332, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ondřej Bojar and Aleš Tamchyna. 2011. Improving translation model by monolingual data. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT '11, pages 330–336, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974. Association for Computational Linguistics.

Mara Chinea-Rios, Álvaro Peris, and Francisco Casacuberta. 2017. Adapting neural machine translation with parallel synthetic data. In *Proceedings of the Second Conference on Machine Translation, Volume 1: Research Papers*, pages 138–147, Copenhagen, Denmark. Association for Computational Linguistics.

Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Josep Maria Crego and Jean Senellart. 2016. Neural machine translation from simplified translations. *CoRR*, abs/1612.06139.

Annad Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156, Copenhagen, Denmark. Association for Computational Linguistics.

Tobias Domhan and Felix Hieber. 2017. Using target-side monolingual data for neural machine translation through multi-task learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1500–1505, Copenhagen, Denmark. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A Simple, Fast, and Effective Reparameterization of IBM Model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia.

Andreas Eisele and Yu Chen. 2010. MultiUN: A Multilingual Corpus from United Nation Documents. In *Proceedings of the Seventh conference on International Language Resources and Evaluation*, pages 2868–2872. European Language Resources Association (ELRA).

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875. Association for Computational Linguistics.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

Anirudh Goyal, Alex Lamb, Ying Zhang, Saizheng Zhang, Aaron C. Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016*, pages 4601–4609, Barcelona, Spain.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, and Yoshua Bengio. 2017. On integrating a language model into neural machine translation. *Comput. Speech Lang.*, 45(C):137–148.

Thanh-Le Ha, Jan Niehues, and Alex Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. In *Proceedings of the 13th International Workshop on Spoken Language Translation (IWSLT 2016)*, Seattle, WA, USA.

Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tieyan Liu, and Wei-Ying Ma. 2016. Dual learning for machine translation. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 820–828. Curran Associates, Inc.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Philipp Koehn. 2005. A parallel corpus for statistical machine translation. In *Proc. MT-Summit*, Phuket, Thailand.

Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical MT. In *Proc. ACL:Systems Demos*, pages 177–180, Prague, Czech Republic.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39, Vancouver. Association for Computational Linguistics.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. *CoRR*, abs/1711.00043.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318, Stroudsburg, PA, USA.

Jaehong Park, Jongyoon Song, and Sungroh Yoon. 2017. Building a neural machine translation system using only synthetic parallel data. *CoRR*, abs/1704.00253.

Alberto Poncelas, Dimitar Shterionov, Andy Way, Gideon Maillette de Buy Wenniger, and Peyman Passban. 2018. Investigating backtranslation in neural machine translation. In *Proceedings of the 21st Annual Conference of the European Association for Machine Translation*, EAMT, Alicante, Spain.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. 2017. Nematus: a toolkit for neural machine translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 65–68, Valencia, Spain. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725. Association for Computational Linguistics.

Raivis Skadiņš, Jörg Tiedemann, Roberts Rozis, and Daiga Deksne. 2014. Billions of parallel words for free: Building and using the eu bookshop corpus. In *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC-2014)*, Reykjavik, Iceland. European Language Resources Association (ELRA).

Miloš Stanojević and Khalil Sima'an. 2014. Fitting sentence level translation evaluation with many dense features. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 202–206, Doha, Qatar. Association for Computational Linguistics.

Weiyue Wang, Jan-Thorsten Peter, Hendrik Rosendahl, and Hermann Ney. 2016. CharacTer: Translation Edit Rate on Character Level. In *Proceedings of the First Conference on Machine Translation*, pages 505–510, Berlin, Germany. Association for Computational Linguistics.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545, Austin, Texas. Association for Computational Linguistics.